



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/723,755

11/26/2003

Keith B. Stobie

13768.459

9956

47973

7590

04/02/2009

WORKMAN NYDEGGER/MICROSOFT

1000 EAGLE GATE TOWER

60 EAST SOUTH TEMPLE

SALT LAKE CITY, UT 84111

EXAMINER

WEI, ZHENG

ART UNIT

PAPER NUMBER

2192

MAIL DATE

DELIVERY MODE

04/02/2009

PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

<b>Office Action Summary</b>	<b>Application No.</b> 10/723,755	<b>Applicant(s)</b> STOBIE ET AL.	
	<b>Examiner</b> ZHENG WEI	<b>Art Unit</b> 2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 22 January 2009.
- 2a) ☐ This action is **FINAL**.                      2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 2-14, 16-20, 23-40 and 42-46 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 2-14, 16-20, 23-40 and 42-46 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All    b) ☐ Some \*    c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- |  |   |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892)                     | 4) <input type="checkbox"/> Interview Summary (PTO-413)           |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____                                      |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)          | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____  | 6) <input type="checkbox"/> Other: _____                          |

**Detailed Action**

***Remarks***

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 01/22/2009 has been entered.
2. This office action is in response to the amendment filed on 01/22/2009.
3. Claim 15 has been cancelled.
4. Claims 45 and 46 have been added
5. Claims 2, 9, 11, 17, 18, 24, 26, 33 and 35 have been amended.
6. Claims 2-14, 16-20, 22-40 and 42-46 remain pending and have been examined.

***Response to Arguments***

7. Applicants' arguments filed on 01/22/2009, in particular on pages 14-15, have been fully considered but they are not persuasive. For example:
  - At page 15, first paragraph, Applicant points out that "Although it may appear that the general concept of providing a matrix of tests and configuration settings is similar to the present invention, Applicant submits that they are very different. Primarily, configuration settings are not similar to verification settings. The configuration settings correspond to software or hardware

settings of the system under test.”. However, Examiner respectfully disagrees. The single test case as recited in the claim and in the current specification actually includes a plurality of sub test cases which covers all different testing functions/instructions for testing the software under test and the verification settings/levels are merely the configurations for different combination of the sub-test cases/functions/instructions. For the testing according to each one of the verification level, the single test case only perform the selected functions/sub test cases (see for example, Fig.1B and Fig.1C and related text). Therefore, as Johnson disclosed, the selected test cases (functions) from the search result of the test case library are organized (verification settings) as a test plan 40 that represents a sequence of test cases to be run on a system under test (see for example, Fig.3, paragraph [0021]) which is similar to the current application wherein after applying configuration/verification settings, only selected test cases/functions are used to perform the test.

- At page 15, second paragraph, Applicant submits that Johnson does not disclose a system that is capable of receiving verification settings that select one of the more than two levels of verification for the test case to perform when executed. However, the Examiner respectfully disagrees. As discussed above, Johnson discloses the test plan (a set of selected test cases) which is selected according to the verification settings/functionalities of the system under test, such as from an evaluation of product knowledge requirement for

- the system (see for example, paragraph [0021] and the example of testing modem operation). Moreover, the test case library with a plurality of test cases is searched and selected (receiving settings) according to the verification settings/configurations and functionalities of the software under test required to be verified (see for example, Fig.2 and Fig.3).
- At page 15, third paragraph, Applicant submits that Johnson does not modify the test cases when different configuration settings are applied. However, the Examiner respectfully disagrees. It can be seen from the discussion above and further in Figure 3 that the modified test case as the Applicant argued is the same as the final test plan in Johnson as shown in Fig.3, including the different test cases which are searched and selected according to the different settings and requirements (modified). Therefore, Johnson discloses all the limitation as the Applicant argued.

### ***Claim Rejections - 35 USC § 102***

8. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Art Unit: 2192

9. Claim 45 is rejected under 35 U.S.C. 102(e) as being anticipated by Johnson (Johnson et al., US 2004/0073890 A1)

Claim 45:

Johnson discloses in a computer system that includes software under test, a method of verifying the software using a single test case (test case library) that can be configured to provide more than two levels of verification (test plan - a set of test cases), the method comprising:

- loading the test case (test case library) into memory of the computer system (see for example, Fig.3, test case library 18, paragraph [0021], “As depicted by FIG.3, test case library 18 is searchable to identify test cases base based on various factors...”), the test case including testing instructions organized to provide more than two levels of verification that can be performed when the test case is executed (see for example, Fig.3, all test cases in test case library; also see paragraph [0010], “each test case having procedures for validating information handling system functionality”);
- receiving verification settings that select one of the more than two levels of verification (a set of test cases in test plan 40) for the test case to perform when executed (Fig.3, test plan 40; also see paragraph [0021], “Selected test cases from the search result are organized as teat plan 40 that represents a sequence of test cases to be run on a system under test); and

Art Unit: 2192

- executing the test plan along with the software such that verification of the software is performed at the level specified by the received verification settings (see for example, Fig.2, project testing 36 step 2 “run tests”. )

### ***Claim Rejections - 35 USC § 103***

10. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

11. Claims 13-17, 25, 39, 44 and 46 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (Johnson et al., US 2004/0073890 A1) in view of Prabhakaran (US 6,859,758) in further view of Bourne (Kelley C. Bourne, Testing Client/Server Systems) and the admitted prior art (APA) of paragraph [0007] of Applicant's background.

Claim 17:

Johnson discloses, in a computer system that includes software under test, a method of verifying the software with one or more tunable test cases that are capable of being set to any of a plurality of verification levels, the method comprising steps for:

- loading one or more test cases that include a plurality of software testing instructions organized as a plurality of verification levels within a verification

- hierarchy, wherein more than two verification levels (test plan) within the verification hierarchy define different amounts of testing for a single test case (test case library) to perform for determining if the software functions as intended when executed (see for example, Figure 2, from step 32, "Test Engineering" to step 34, "Project Engineering", "Test Cases" and related text);
- receiving verification setting instructions for one or more desired verification levels from within the verification hierarchy for use in testing the software, wherein the received verification setting instructions select the one or more desired verification levels from a group of more than two verification levels (test plan – a set of test cases) that include at least first and second verification levels, (see for example, Figure 2, step about passing "Configuration Information" to step 34, "Project Engineering" and related text); and
  - testing the software at the one or more desired verification levels, which include at least one of the first and second verification levels, by running the one or more test cases that include the plurality of software testing instructions that correspond to the one or more desired verification levels (see for example, Figure 2, step 36 "Project Testing" and related detailed steps and text).

But does not explicitly disclose wherein selection of the first verification level causes the one or more test cases to be run during testing and which includes invoking an insert record object to determine if the invocation of the insert record



object results in a system crash and while refraining from producing any recorded output, and wherein selection of the second verification level, which is distinguished from the first verification level causes the one or more test cases to be run with different instructions and invoke an insert record object and to additionally verify through recorded output that a record corresponding to the insert record object was properly inserted and presented when the one or more test cases corresponding to the second verification level are run and such that the recorded output which is produced in response to the one or more test cases being run following the selection of the second verification level is refrained from being produced in response to the one or more test cases being run following the selection of the first verification level.

However, Prabhakaran in the same analogous art of software testing discloses a method and system for stress testing database storage wherein selection of the stress test causes the one or more test cases to invoke an insert record object and to additionally verify through recorded output that a record corresponding to the insert record object was properly inserted and presented (see for example, col.7, lines 36-44, "data be stored in a database", "notifies stressing software 440 that the information has been successfully stored in the database"; also see Fig.4, item 450 "Monitor software" and related text). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to include Prabhakaran's test case into Johnson's test management system. One would have been motivated to do so to simplify test case and

configuration re-use as suggested by Johnson (see for example, ABSTRACT, “simply test case and configuration re-use”).

Prabhakaran further discloses selection of the test case causes the one or more test cases to be run during testing and which includes invoking an insert record object to determine if the invocation of the insert record object results in a system crash (see for example, col.7, lines 45-58, “maximum rate of operations to enterprise storage system 410 is achieved”).

But Johnson and Prabhakaran do not explicitly disclose the limitation about refraining from producing any recorded output.

However, APA discloses a concept of stress test that simply ignores any testing output if system doesn't crash when the insert record object is run (see for example, paragraph [0009]) and Bourne also discloses detailed information about “stress test” is that “stress testing determines if the system will break down or otherwise malfunction when it is being overloaded” (see for example, p.356, section 11.1 Stress Testing, second paragraph) Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify and run Prabhakaran and Johnson's test case for only focusing on the condition of system crash without producing any recorded output. Because, the test output is not important and the system does not analyze the output as pointed out by APA (see for example, page 4, paragraph [0009], “the system does not analyze the output”). One would have been motivated to do so to make test procedure more efficient as suggest by APA (see for example, paragraph

Art Unit: 2192

[0009], "Producing the output in the first place, however, impacts the system, so it would be better not to produce it in the first place.") and also suggested by the Bourne's purpose of the stress testing to determine system crash as addressed above.

Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to combine Prabhakaran, APA and Bourne's teachings into Johnson's system and using Johnson's test case management and customization feature to customize Prabhakaran's stress test case to test database storage using first/second verification level as addressed above according to different requirement functional test (successfully stored in database) or system crash test (maximum rate of operations) as disclosed above.

Claim 13:

Johnson further discloses the method of claim 17, wherein at least a portion of at least one of the plurality of software instructions determines that software information is available and uses the information for troubleshooting the software if it is determined that the software does not function as intended when executed (see for example, Figure 2, step 3-5 of "Project Testing 36", "Record Results", "Report Issues", "Provide Test Case Feedback when necessary" and related text).

Art Unit: 2192

Claim 14:

Johnson also discloses the method of claim 13, wherein the software information available is debug information (see for example, Figure 2, step 3-5 of “Project Testing 36”, “Provide Test Case Feedback when necessary” and related text, also see, p.3, paragraph [0023], “As tests are run and results recorded, report are issued to test engineering for tracking test progress and adapting tests with feedback”).

Claim 16:

Johnson further discloses the method of claim 17, wherein the portion of the one or more test cases that corresponds to the one or more desired verification levels produces one or more test outputs for verifying the software (see for example, Figure 2, step 3-5 of “Project Testing 36”, “Record Results”, “Report Issues”, “Provide Test Case Feedback when necessary” and related text).

Claim 25

Claim 25 is a computer program product version of claimed method in claim 17 above, wherein all claimed limitations have been address and/or set forth above by Johnson and APA. Therefore, as the references teach all the limitation, they also teach the limitations of claim 25. Thus, it also would have been obvious.

Claim 39:

Art Unit: 2192

Johnson, Prabhakaran, APA and Bourne's disclose the method of claim 25, but does not disclose wherein the portion of the one or more test cases that corresponds to the one or more desired verification levels does not produce any testing output.

However, APA discloses the stress test that simply ignores any testing output if system doesn't crash when the insert record object is run (see for example, paragraph [0009]). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify and run Johnson's test case for simple stress tests without producing any output. Because, the test output is not important and the system does not analyze the output as pointed out by APA (see for example, page 4, paragraph [0009], "the system does not analyze the output"). One would have been motivated to do so to make test procedure more efficient as suggest by APA (see for example, paragraph [0009], "... , so it would be better not to produce it in the first place.")

Claim 44:

Johnson, Prabhakaran, APA and Bourne's disclose the method as recited in claim 17 Johnson further discloses wherein the method further includes upon detecting and adverse or unexpected result form testing the software, determining of which of the test cases has caused the adverse or unexpected result is accomplished by isolating the plurality of test case within the test group and running each of the isolated test cased individually (see for example, p.2-3, paragraph [0020], "the number of tests and results for tests performed under a

Art Unit: 2192

predetermined test case or configuration is traceable to view how many times the test case or configuration was used, passed or failed”; also see paragraph [0023], “As tests are run and results recorded, reports are issued to test engineering for tracking test progress and adapting test with feedback”; also see paragraph [0023], “After repetitions of the test cases, test engineers may view results to update test case where testing failures are encouraged by test case faults...”).

Claim 46:

Johnson discloses in a computer system that includes a software under test that is to be tested as addressed in claim 45 (see for example, rejection in claim 45 above), but does not explicitly disclose the software under test is a database and the verification levels of the database insertion comprise: a first verification level that causes the test case to produce no output when executed; a second verification level that causes the test case to produce an output that is analyzed to determine whether the insert record object functioned correctly; a third verification level that causes the test case to check the database to verify that the insert record object did not insert a record twice or over-write another record unintentionally; and a fourth verification level that causes the test case to verify that memory and disk space was used as expected and that event log messages were as expected

However, Prabhakaran in the same analogous art of software testing discloses a method and system for stress testing database storage wherein selection of the

stress test causes the one or more test cases to invoke an insert record object and to additionally verify through recorded output that a record corresponding to the insert record object was properly inserted and functioned correctly (see for example, col.7, lines 36-44, "data be stored in a database", "notifies stressing software 440 that the information has been successfully stored in the database"; also see Fig.4, item 450 "Monitor software" and related text). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to include Prabhakaran's test case into Johnson's test management system to verify the insert record functioned correctly including the insert record object did not insert a record twice or over-write others. One would have been motivated to do so to simplify test case and configuration re-use as suggested by Johnson (see for example, ABSTRACT, "simply test case and configuration re-use").

Prabhakaran further discloses selection of the test case causes the one or more test cases to be run during testing and which includes invoking an insert record object to determine if the invocation of the insert record object results in a system crash (see for example, col.7, lines 45-58, "maximum rate of operations to enterprise storage system 410 is achieved").

But Johnson and Prabhakaran do not explicitly disclose the limitation about causing the test case to produce no output and to verify memory/disk space/event message.

Art Unit: 2192

However, APA discloses a concept of stress test that simply ignores any testing output if system doesn't crash when the insert record object is run (see for example, paragraph [0009]) and Bourne further discloses detailed information about "stress test" is that "stress testing determines if the system will break down or otherwise malfunction when it is being overloaded (see for example, p.356, section 11.1 Stress Testing, second paragraph). Moreover, Bourne also discloses the stress testing must be specifically designed to test/verify memory and disk space (see for example, p.356, last paragraph, "Memory", "Disk space") Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify and run Prabhakaran and Johnson's test case for only focusing on the condition of system crash without producing any output. Because, the test output is not important and the system does not analyze the output as pointed out by APA (see for example, page 4, paragraph [0009], "the system does not analyze the output") and also can be used to verify the memory and disk space resource to determine if the system will break down (see for example, p.356. last two paragraphs). One would have been motivated to do so to make test procedure more efficient as suggest by APA (see for example, paragraph [0009], "Producing the output in the first place, however, impacts the system, so it would be better not to produce it in the first place.") and also suggested by the Bourne's purpose of the stress testing to determine system break down as addressed above.



Art Unit: 2192

Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to combine Prabhakaran, APA and Bourne's teachings into Johnson's system and using Johnson's test case management and customization feature to customize Prabhakaran's stress test case to test database storage using first/second/third/fourth verification level as addressed above according to different requirement functional test (successfully stored in database) or system break down test (maximum rate of operations) or system running out of recourse including memory and disk space as disclosed above.

12. Claims 2-12 and 23-24, 18-20, 26-38, 40 and 42-43 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (Johnson et al., US 2004/0073890 A1) in view of Prabhakaran (US 6,859,758) in further view of Bourne (Kelley C. Bourne, Testing Client/Server Systems) and the admitted prior art (APA) of paragraph [0007] of Applicant's background and in further view of Ruffolo (Ruffolo et al., US 2003/0196190 A1).

Claim 2:

Johnson , Prabhakaran, APA and Bourne disclose the method of claim 17, wherein a first test case from the more than two test cases (test plan) is part of a first test group, the first test group including one or more software testing instructions organized as one or more verification levels within the verification hierarchy, and wherein the verification settings (configurations) that define one or more desired verification levels (Test Iteration) for the first test group (Test Plan)

Art Unit: 2192

(see for example, Figure 1B, element 30, "Configurations", element 28, "Test Plan", "Test Case", element 26 "Test Iteration" and related text).

But do not disclose the verification settings defining a desired verification level for the one or more test cases. However, Ruffolo in the same analogous art of test case generation discloses building different verification level (test items) of test case based on verification settings (distribution list) (see for example, Fig.4, step S406-S412 and relate text). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to define the verification settings for the test case in the configuration file to further customize the verification level of each test case. One would have been motivated to do so to customize each test case for the project as suggested by Johnson (see for example, Figure 2, step 2a of "Project Engineering 34" – "Customize Test Cases for the project").

Claim 3:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 2, Johnson further discloses the method comprising acts of:

- identifying a portion of the one or more software testing instructions within the first test group that corresponds to the one or more desired verification levels (see for example, p.1, paragraph [0010], "A test iteration engine aligns a test case or set of test cases with a configuration to present a matrix view of one or more test cells that guide testing of an information handling system having

Art Unit: 2192

- the identified configuration, also see Figure 1B, element 26, “Test Iteration”, element 28, “Test Plan”, element 30, “Configurations” and related text)
- running a portion of the first test group that corresponds to the one or more desired verification levels (see for example, Figure 2, step 36 “Project Testing” and related detailed steps and text).

Claim 4:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 3, Johnson also discloses, wherein the verification settings (configurations) define a single desired verification level for the first test case and the first test group (see for example, Figure 1B, “Configuration B” of element 30 “Configurations”, using single configuration to cover all test cases in “Test Plan 28”, also see related text descriptions).

Claims 5 and 7:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 3, but do not explicitly disclose that the verification settings defined verification level for the first/second test cases is different from a desired verification level for the first test group.

However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the

Art Unit: 2192

first/second test cases and test group are different, because each test groups comprises one or more test cases, each test cases can be customized to different verification level to test different degree or portion of software component based on different configurations as discussed above. Therefore, verification levels of the test case and test group can be different.

Claim 6:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 4, but do not explicitly disclose that the verification settings defined verification level for the second test case are different from a desired verification level for the first test group.

However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the first/second test cases and test group are different, because each test groups comprises one or more test cases, each test cases can be customized to different verification level to test different degree or portion of software component based on different configurations as discussed above. Therefore, verification levels of the test case and test group can be different.

Claim 8:

Art Unit: 2192

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 7, but do not explicitly disclose that the verification settings defined verification level for the first/second test cases is different.

However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the first/second test cases could be different. Because each test cases can be customized to different verification level to test different degree or portion of software component based on different configurations as discussed above. Therefore, verification levels of the test cases can be different.

Claim 9:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 3, Johnson further discloses wherein a second test case from the more than two test cases is part of the first test group, and wherein third and fourth test cases from the one or more test cases are part of a second test group, the second test group including one or more software testing instructions organized as one or more verification levels within the verification hierarchy, and wherein the verification settings that define the one or more desired verification levels for the one or more test cases also define one or more desired verification levels for the second test group, the method further comprising acts of:

- identifying a portion of the one or more software testing instructions within the second test group that corresponds to the one or more desired verification

Art Unit: 2192

levels (see for example, p.1, paragraph [0010], “A test iteration engine aligns a test case or set of test cases with a configuration to present a matrix view of one or more test cells that guide testing of an information handling system having the identified configuration, also see Figure 1B, element 26, “Test Iteration”, element 28, “Test Plan”, element 30, “Configurations” and related text); and

- running a portion of the second test group that corresponds to the one or more desired verification levels (see for example, Figure 2, step 36 “Project Testing” and related detailed steps and text).

Claim 10:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 9, but do not explicitly disclose that the verification settings defined verification level for the first/second/third/fourth test cases, the first test group and second test group are different.

However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the test cases and test groups can be set to different verification levels, because each test groups comprises one or more test cases, each test cases can be customized to different verification level to test different degree or portion of

Art Unit: 2192

software component based on different configurations as discussed above.

Therefore, verification levels of the test cases and test groups can be different.

Claim 11:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 10, Johnson further discloses wherein the first and second test groups are part of a third test group, the third test group including more than two software testing instructions organized as one or more verification levels within the verification hierarchy, and wherein the verification settings that define the one or more desired verification levels for the one or more test cases also define one or more desired verification levels for the third test group, the method further comprising acts of:

- identifying a portion of the one or more software testing instructions within the second test group that corresponds to the one or more desired verification levels (see for example, p.1, paragraph [0010], “A test iteration engine aligns a test case or set of test cases with a configuration to present a matrix view of one or more test cells that guide testing of an information handling system having the identified configuration, also see Figure 1B, element 26, “Test Iteration”, element 28, “Test Plan”, element 30, “Configurations” and related text); and

Art Unit: 2192

- running a portion of the second test group that corresponds to the one or more desired verification levels (see for example, Figure 2, step 36 “Project Testing” and related detailed steps and text).

Claim 12:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 9, but do not explicitly disclose that the verification settings define a desired verification level for the third test group different from each of the first test case, the second test case, the third test case, the fourth test case, the first test group and the second test group.

However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the test cases and test groups can be set to different verification levels, because each test groups comprises one or more test cases, each test cases can be customized to different verification level to test different degree or portion of software component based on different configurations as discussed above.

Therefore, verification levels of the test cases and test groups can be different.

Claims 23-24:

Claims 23-24 are a computer program product version of claimed method, wherein all claimed limitations have been address and/or set forth above in



Art Unit: 2192

claims 2-14 and 16-17. Therefore, as the references teach all the limitation of claims 2-14 and 16-17, they also teach the limitations of claims 23-24 respectively. Thus, they also would have been obvious.

Claim 18:

Johnson, Prabhakaran, APA and Bourne discloses the method of claim 17, wherein a first test case from the one or more test cases is part of a first or a second test group, the first test group including one or more software testing instructions organized as more than two verification levels within the verification hierarchy, further comprising acts of:

- identifying a portion of the one or more software testing instructions within the first test group that corresponds to the one or more desired verification levels (see for example, p.1, paragraph [0010], “A test iteration engine aligns a test case or set of test cases with a configuration to present a matrix view of one or more test cells that guide testing of an information handling system having the identified configuration, also see Figure 1B, element 26, “Test Iteration”, element 28, “Test Plan”, element 30, “Configurations” and related text); and
- running a portion of the first test group that corresponds to the one or more desired verification levels (see for example, Figure 2, step 36 “Project Testing” and related detailed steps and text)

But does not disclose the verification settings defining a desired verification level for the one or more test cases. However, Ruffolo in the same analogous art of

Art Unit: 2192

test case generation discloses building different verification level (test items) of test case based on verification settings (distribution list) (see for example, Fig.4, step S406-S412 and relate text). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to define the verification settings for the test case in the configuration file to further customize the verification level of each test case. One would have been motivated to do so to customize each test case for the project as suggested by Johnson (see for example, Figure 2, step 2a of “Project Engineering 34” – “Customize Test Cases for the project”).

Johnson and Ruffolo also do not explicitly disclose the verification level for the first test case is different form a desired verification level for the first test group. However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the first test case and first test group could be different. Because each test groups can be customized to different verification level to test different degree or portion of software component based on different configurations as discussed above. Therefore, verification levels of the test cases and test group can be different.

Claim 19:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 18, wherein a second test case from the one or more test cases is part of the first test group, but do not explicitly disclose the verification level for the second test case is different form a desired verification level for the first test group. However,

Art Unit: 2192

it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the first test case and first test group could be different. Because each test groups can be customized to different verification level to test different degree or portion of software component based on different configurations as discussed above. Therefore, verification levels of the test cases and test group can be different.

Claim 20:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 19, Johnson further discloses wherein verification setting instructions for the desired verification levels define a single verification level for the first and second test cases (see for example, Figure 1B, "Configuration B" of element 30 "Configurations", using single configuration to cover all test cases in "Test Plan 28", also see related text descriptions).

Claims 26-38 and 40:

Claims 26- 38 and 40 are a computer program product version of claimed method in claims 17-20 and 25 above, wherein all claimed limitations have been address and/or set forth above by Johnson and Ruffolo. Therefore, as the references teach all the limitation, they also teach the limitations of claims 25-38 and 40 respectively. Thus, they also would have been obvious.

Claims 42:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose a [The] method as recited in claim 17,

but do not explicitly disclose wherein selection of a third verification level causes verification of the record being inserted as well as verification that the record was only inserted a single time and wherein testing of the software includes running the third verification level. However, Ruffolo in the same analogous art of test case generation discloses building different verification level (test items) of test case based on verification settings (distribution list) (see for example, Fig.4, step S406-S412 and relate text). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to define the verification settings for the test case in the configuration file to further customize the verification level of each test case including just testing a single time insertion. One would have been motivated to do so to customize each test case for the project as suggested by Johnson (see for example, Figure 2, step 2a of “Project Engineering 34” – “Customize Test Cases for the project”).

Claim 43:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose a [The] method as recited in claim 17, but do not explicitly disclose wherein selection of a third verification level causes verification of the record being inserted as well as

Art Unit: 2192

verification that the record was inserted without overwriting another record and wherein testing of the software including running the third verification level.

However, Ruffolo in the same analogous art of test case generation discloses building different verification level (test items) of test case based on verification settings (distribution list) (see for example, Fig.4, step S406-S412 and relate text). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to define the verification settings for the test case in the configuration file to further customize the verification level of each test case including just testing a single time insertion. One would have been motivated to do so to customize each test case for the project as suggested by Johnson (see for example, Figure 2, step 2a of "Project Engineering 34" – "Customize Test Cases for the project").

### ***Conclusion***

13. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.
14. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Zheng Wei whose telephone number is (571) 270-1059 and Fax number is (571) 270-2059. The examiner can normally be reached on Monday-Thursday 8:00-15:00.

Art Unit: 2192

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature of relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571- 272-1000.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Z. W./  
Examiner, Art Unit 2192

/Tuan Q. Dam/  
Supervisory Patent Examiner, Art Unit 2192